Benutzerhandbuch

HTS

Hengstler Terminal Server

für Windows

HENGSTLER GmbH
Postfach 11 51

78 550 Aldingen

Sachnr.: 2 723 101
Änderungsstand: 10.02.98

Technische Änderungen und Verbesserungen, die dem Fortschritt unserer Geräte dienen, behalten wir uns vor.

Inhalt

Einführung

Dieses Benutzerhandbuch beschreibt die Installation und Inbetriebnahme des HTS sowie dessen Schnittstellen zu den Anwendungsprogrammen und ist anwendbar für:

- **HTS für Windows 3.x**
  Version: 1.47 und höher
  Sach-Nr. 0 723 165
- **HTS für Windows 95 und NT**
  Version: 1.47 und höher
  Sach-Nr. 0 723 167

Beiden Varianten verhalten sich hinsichtlich der Programmausführung genau gleich, lediglich die Installations-prozedur unterscheidet sich. *Nicht* enthalten sind Beschreibungen der internen Datenstrukturen von Zählern, RTC, Terminals oder LTA. Hierzu sei auf die jeweilige Gerätedokumentationen verwiesen.

Aufgaben des HTS

Das nachfolgende Schema gibt einen groben Überblick über die Funktion des HTS im PC:

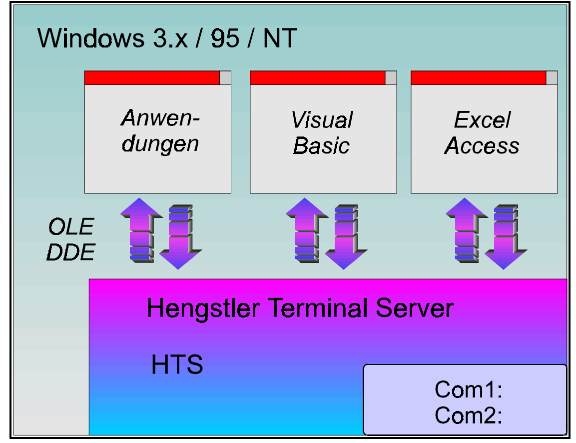


Abb. 1: HTS mit OLE und DDE

DDE: Dynamic Data Exchange
Ein sogenannter DDE-Server sorgt für die Ermittlung der variablen Daten (hier die Zählerregister). Andere Anwendungen, sogenannte DDE-Clients, tauschen Daten mit dem DDE-Server über vereinbarte Mechanismen aus. Dabei dient globaler Speicherbereich in Windows zur Pufferung der Daten. Dieser Speicherbereich ist jedoch nur durch 16-bit-Programme erreichbar.

OLE: Object Linking and Embedding
Die Kommunikation basiert hierbei nicht auf reinen Daten, sondern vielmehr auf dem Einbinden von Funktionen des OLE-Serversprogramms, mit denen die gewünschten Daten ermittelt werden.

DDE und OLE stehen gleichzeitig zur Verfügung

Aufgaben des HTS	Beschreibung
Installationsprogramm	• Installationsprozedur • Programmgruppe und -Icon
Kommunikationsprozeß zu den PC-Schnittstellen COMx	• Masterfunktion für Datenprotokoll *Hengstler TP3* • Modembetrieb • direkte Kommunikation bei Punkt-zu-Punkt • indirekte Kommunikation über Buskonverter RTC
Benutzeroberfläche	• Konfiguration des HTS • Konfiguration der angeschlossenen Geräte • Statistik • Diagnosefunktionen • Fehlermeldungen
Schnittstelle für die Kom-munikation mit Anwendungen	• DDE (Dynamic Data Exchange) für 16-Bit-Anwendungen • OLE (Object Linking and Embedding) für 16- und 32-Bit-Anwendungen

Tabelle 1: Aufgaben des HTS

Kompatibilität

Nachfolgende Tabelle zeigt das Zusammenspiel bei der Kommunikation zwischen HTS und Anwendungen auf den unterschiedlichen Betriebssystemen. Zu beachte ist, daß DDE nur im 16-bit-Adressraum definiert ist und daher nur mit 16-bit-Anwendungen möglich ist.

Plattform	Kommunikationsform
Windows 3.1 / Windows 3.11	OLE (16 bit) DDE (16 bit)
Windows NT / Windows 95	OLE (16 bit oder 32 bit) DDE (nur für 16-Bit-Programme)

Tabelle 2: HTS Kommunikation

Topologie

Zwei grundsätzliche Vernetzungsarten stehen zur Verfügung:

- **Punkt-zu-Punkt-Verbindung**
  Ein einzelner Zähler wird direkt am PC angeschlossen. Der Zähler besitzt eine RS232-Schnittstelle mit TP3-Protokoll.
- **Verbund von bis zu 31 Zählern**
  Die Geräte verfügen jeweils über eine RS485-Schnittstelle mit TP3-Protokoll und werden am Zweidraht-Bus angeschlossen. Der Buskonverter RTC sorgt für die Umsetzung von RS485 auf RS232 zum Anschluß an den PC. HTS und RTC sind optimal aufeinander abgestimmt.

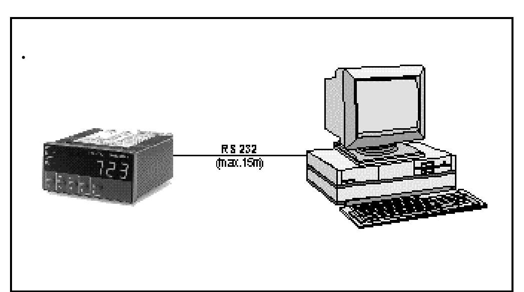


Abb. 2: Punkt-zu-Punkt-Verbindung

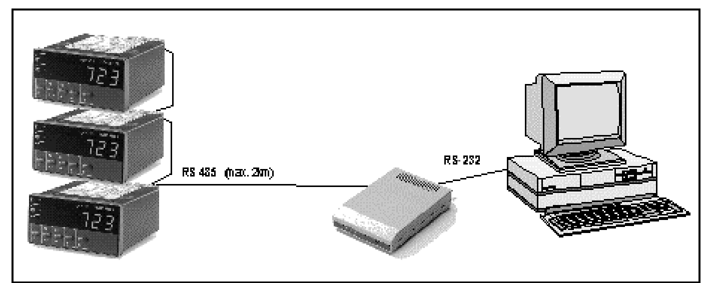


Abb. 3: Zähler im Netzverbund

An einer COM-Schnittstelle am PC sind vier grundsätzliche Anordnungen möglich:

S)	**1x Zähler**	**RS232**			**<—>**		**RS232**	**COMx**
S1)	**1x Zähler**	**RS232**		**Modem**	**<—>**	**Modem**	**RS232**	**COMx**
N)	**Zähler 1** **...** **Zähler 31**	**RS485** **...** **RS485**	**RTC**		**<—>**		**RS232**	**COMx**
N1)	**Zähler 1** **...** **Zähler 31**	**RS485** **...** **RS485**	**RTC**	**Modem**	**<—>**	**Modem**	**RS232**	**COMx**

Tabelle 3: Vier grundsätzliche Anordnungen

Installation

Systemvoraussetzungen

Folgende Mindestanforderungen sollten erfüllt sein:

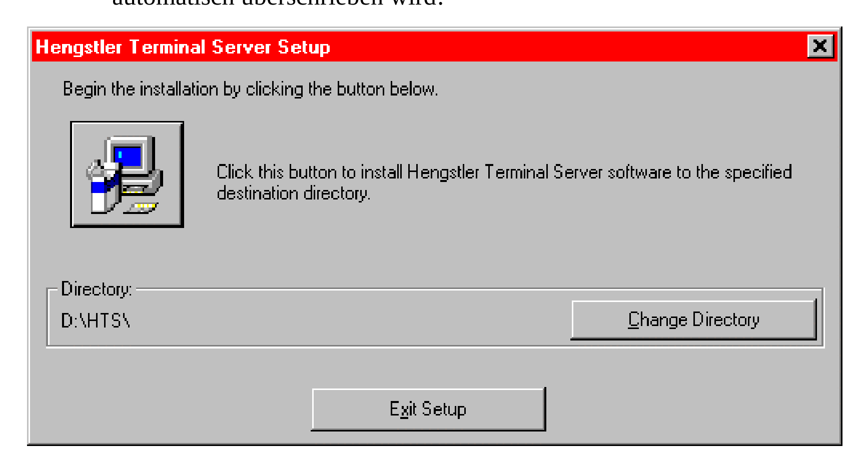
- IBM-kompatibler PC
- mindestens 386er
- Arbeitsspeicher mindestens 4 MB
- Windows 3.1 / 3.11 oder Windows 95 oder Windows NT
- eine COM-Schnittstelle, die korrekt in Windows konfiguriert und ansprechbar ist
- ca. 2 MB freien Festplattenspeicher

Setup

Die Installation erfolgt von Diskette durch Starten des Programmes ***setup.exe***. Sie können das angebotene Zielverzeichnis ändern und dann mit dem großen Button die eigentliche Installation beginnen. Setup kopiert alle erforderlichen Dateien auf die Festplatte und legt die Programmgruppe "Hengstler Terminal Server" und ein Startsymbol an. Darüberhinaus nimmt Setup auch den Eintrag des HTS in die Registrierdatenbank (regedit) von Windows vor, der für die OLE-Schnittstelle benötigt wird.



Wenn Sie bereits eine ältere Version des HTS installiert haben, müssen Sie vor der Neuinstallation ***hentmsrv.exe*** auf dem bisherigen Verzeichnis löschen, da es nicht automatisch überschrieben wird!



Setup meldet die erfolgreich durchgeführte Installaton mit folgender Dialogbox:



Autostart

Um HTS bei jedem Neustart von Windows automatisch zu starten, tragen Sie HTS in die Programmgruppe "Autostart" ein.

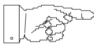
In Windows 3.1x und NT:
1. Programmgruppe "Autostart" öffnen
2. Programmgruppe "Hengstler Terminal Server" öffnen
3. Beide Gruppen so ausrichten, daß sie gleichzeitig sichtbar sind
4. Programmsymbol "Hengstler Terminal Server" einmal anklicken, STRG-Taste gedrückt halten und das Symbol in die Programmgruppe "Autostart" ziehen und loslassen.

In Windows 95:
1. Mit rechter Maustaste auf START klicken und Öffnen anklicken
2. In das Verzeichnis "Programme" und dort nach "Hengstler Terminal Server" verzweigen
3. Programmsymbol "Hengstler Terminal Server" einmal anklicken und mit Strg-C Kopieren
4. Eine Verzeichnisebene zurück und nach "Autostart" verzweigen
5. Mit Strg-V kopieren Sie die Verknüpfung in diese Gruppe

Soll kein Autostart mehr erfolgen, so ist das Programmsymbol aus "Autostart" zu löschen.

Inbetriebnahme

HTS ist nach der Installation vorkonfiguriert wie in nachfolgender Übersicht dargestellt. Wenn Ihre Systemanordnung mit der Voreinstellung übereinstimmt und der Zähler auf die physikalische Adresse 1 programmiert ist (vgl. F-Codes), dann brauchen Sie grundsätzlich keine Änderungen an den Einstellungen vornehmen, um HTS mit Ihrer Hardware in Betrieb zu nehmen.



Es wird empfohlen, alle beschriebenen Schritte durchzuführen.

Gerät	Terminal-Nummer	Terminal-Name	physik. Adresse	Terminal-Klasse	Daten-übertrg.	COM-Port
Zähler	0001	Zaehler1	1	0005-Zähler	SCC1: 4800 Bd	COM1
RTC	0000	RTC	0	0000-RTC	SCC1: 9600 Bd SCC2: 4800 Bd	COM1

Tabelle 4: Voreinstellungen von HTS nach der Installation

Entfernen Sie zunächst alle angeschlossenen Zähler und ggfs. den RTC von der oder den COM-Schnittstellen. Die Inbetriebnahme gestaltet sich wie folgt:

1. **Erstes Starten von HTS**
   Schnittstellen noch nicht beschalten, keine Zähler anschließen
   automatische Registrierung des OLE beim ersten Aufruf

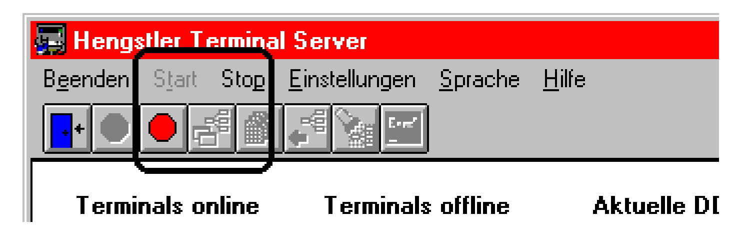
2. **allgemeine Treiber-Einstellungen von HTS vornehmen**
   Menü: Einstellungen -> Treiber-Einstellungen

3. **Einrichten des/der Zähler und der Schnittstellen entsprechend der Systemanordnung**
   Festlegen der Systemanordnung
   Menü: Einstellungen -> Terminal-Klassen
   Menü: Einstellungen -> Terminal-Konfiguration

4. **Starten von HTS**
   Menü: Start
   Bewerten der Statusanzeige der einzelnen Bildschirmfelder von HTS
   Übersicht der jeweiligen Einstellungen
   Meldungen und Statistik bewerten

Erstes Starten

Die erste Inbetriebnahme des HTS für das Einstellen der Treiberparameter erfolgt ohne angeschlossene Zähler- und RTC-Hardware. Starten Sie HTS durch Doppelklicken auf das Programmsymbol in der entsprechenden Programmgruppe bzw. in Windows 95 über Start -> Programme -> Hengstler Terminal Server (im weiteren Verlauf dieses Handbuches wird nicht mehr darauf eingegangen, wie HTS zu starten ist). Ignorieren Sie zunächst Meldungen während der Startphase.



Setzen Sie den HTS auf offline durch anklicken des roten Stopsymbols oder des Menüpunktes ***Stop***.

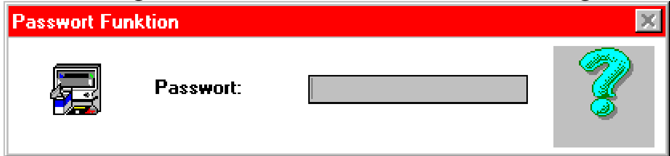


Einstellungen am Treiber können nur vorgenommen werden, solange der HTS auf Stop steht.



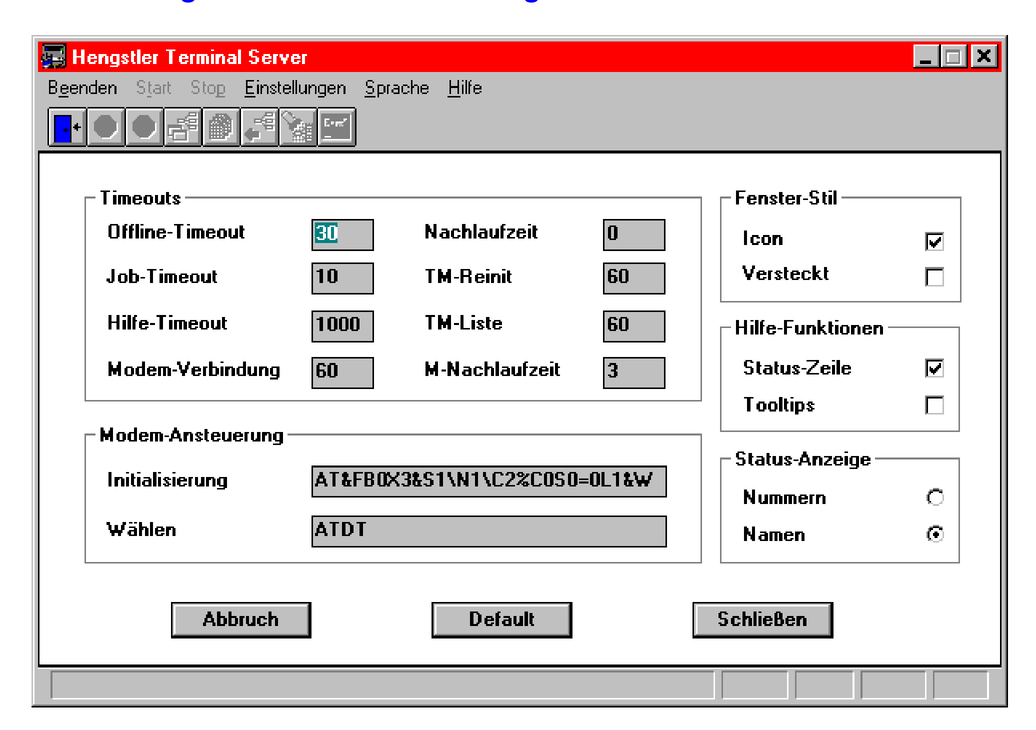
HTS verwendet den Begriff **Terminal** (Terminal = Endstelle) für alle anschließbaren Geräte, also auch für Zähler. Lassen Sie sich daher nicht verwirren. Von **Öffnen** ist die Rede, wenn HTS die Verbindung zu einem Terminal aufnimmt.

Bevor Sie eines der Menüs unter ***Einstellungen*** zum ersten mal nach Programmstart aufrufen, müssen Sie das Passwort eingeben, das Ihnen bei Erhalt der Diskette mitgeteilt worden ist.



Das Passwort lautet: „HTS1996“

Einstellungen - Treiber-Einstellungen





Die in ( ) angegebenen Werte sind die Standardvorgabewerte (Default).
Mit ⇒ markierte Werte können geändert werden. Alle anderen Einstellungen sollten nur von erfahrenen Benutzern verändert werden!

Offline-Timeout (30)
Anzahl Pollzyklen (Abfragevorgang innerhalb des Protokolls), nach denen ein nicht antwortender RTC offline gemeldet wird.

Job-Timeout (10 Sekunden)
Zeit in Sekunden, nach der das erste Antwortpaket auf einen angestoßenen Job empfangen werden muß, bevor der Job verworfen und mit entsprechendem Fehler quittiert wird.

⇒Hilfe-Timeout (1000 Millisekunden)
Hier kann eingestellt werden mit welcher Verzögerung die gelben Tooltips auf dem Bildschirm erscheinen. Die Angabe erfolgt in Millisekunden. Zur Zeit ohne Auswirkung, da Tooltips in dieser Version nicht realisiert sind. Stattdessen erscheint in der Statuszeile der Hilfstext des gerade überstrichenen Buttons.

⇒Modem-Verbindung (60 Sekunden)
Maximale Zeit in Sekunden, die ein Modem-Verbindungsaufbau dauern darf. Wenn nach der eingestellten Zeit keine Modem-Verbindung zustande kommt, wird der Verbindungsaufbau abgebrochen und eine entsprechende Fehlermeldung erzeugt.

⇒Nachlaufzeit (60)
Nachlaufzeit des Servers in Sekunden. Bei einem Eintrag größer 0 beendet sich HTS selbsttätig nach der angegeben Zeit, nach der letzten aktiven DDE- oder OLE-Verbindung. Der Wert 0 verhindert das selbsttätige Beenden des Servers. Dies trifft nicht zu, solange der Server auf Stop steht.

TM-Reinit (60 Sekunden)
Wartezeit in Sekunden, nach der erneut versucht wird, die Verbindung zu einem Terminal aufzunehmen, dessen Initialisierungsdaten nicht gelesen werden konnten. Dieser Parameter ist für Zähler und RTC nicht von Bedeutung.

TM-Liste (60 Sekunden)
Intervallzeit in Sekunden, nach der die Terminal-Liste vom RTC gelesen werden soll. Zusätzlich wird immer dann, wenn vom RTC ein entsprechendes Meldekommando geschickt wurde, die Terminal-Liste neu gelesen. Dieser Parameter ist für Zähler und RTC nicht von Bedeutung.

⇒M-Nachlaufzeit (3 Sekunden)
Modem-Nachlaufzeit in Sekunden. Dabei handelt es sich um die Zeit, die eine Modem-Verbindung nach dem Schließen des Terminals noch offengehalten wird, um bei einem erneuten Öffnen des Terminals nicht neu aufgebaut werden zu müssen.

⇒Status-Zeile
Dieser Parameter gibt an ob in der Fußzeile für bestimmte Masken Hilfetexte angezeigt werden sollen.

⇒Tooltips
Gibt an, ob Hilfetext in bestimmten Masken in Form von sprechblasenartigen Hilfstexten (=Tooltips) angezeigt werden sollen. In dieser Version ohne Auswirkung (siehe auch unter "*Hilfe-Timeout*").

⇒Icon
Gibt an, daß der Server nach seinem Start nicht als Vollbild, sondern als Icon angezeigt werden soll. Mit Alt-Tab erscheint dennoch das Icon in der Taskleiste und signalisiert, daß der Server läuft.

Hidden
Gibt an, daß der Server nach seinem Start nicht als Vollbild, sondern unsichtbar im Hintergrund läuft. Dieser Parameter übersteuert den Icon-Parameter. Der Server erscheint auch ***nicht*** in der Windows-Taskleiste.



Das Setzen dieses Parameters kann nicht innerhalb einer Maske rückgängig gemacht werden, da keine Maske mehr zugänglich ist. Ebenso kann in diesem Betriebsmodus der Server nur noch durch Definition einer Nachlaufzeit oder das Beenden von Windows gestoppt werden, da die entsprechenden Icons nicht mehr erreichbar sind.
Sie können die normale Betriebsart wiederherstellen durch Setzen von ***WindowMode=0*** unter ***[Global]*** in ***hentmsrv.ini*** mittels eines Editors.

⇒Status-Anzeige Nummern / Namen
Gibt an, ob die Geräte in den Spalten Terminals online bzw. Terminals offline mit ihrer Terminal-Nummer oder ihrem Namen angegeben werden.

⇒Initialisierung / Wählen
Dieser String wird vor dem Verbindungsaufbau an das Modem gesendet, falls die Verbindung über Modem läuft (vergleiche Menüpunkt Terminal-Konfiguration). Wählen ist der Präfix-String, der vor der eigentlichen Telefon-nummer an das Modem gesendet wird. Hier können Sie auch die für Ihre Telefonanlage relevanten Zeichen für Amtsholung usw. eintragen. Passen Sie ggfs. die Strings Ihrem Modem an.
Der Treiber arbeitet mit Modemantworten im Klartext, deshalb muß im Initstring ein V1 stehen (V0 bedeutet Kurz-form als Ziffer). Register S0 (Anzahl Klingelimpulse vor Rufannahme) muß auf mindestens 1 gesetzt werden. S0=0 bedeutet keine Anrufannahme. Mit der Steuerleitung DTR wird der Ruf unterbrochen bzw. aufgenommen, was mit &D2 einzustellen ist. Echomodus muß durch E0 ausgeschaltet werden. Die Anzahl der Rückmeldungen muß auf 10 begrenzt werden, was mit X eingestellt wird. Mit X4 erfolgt die Begrenzung auf 10. Die meisten modernen Modems unterbrechen nach einer bestimmten Zeit ohne Datenverkehr automatisch die Verbindung. Dies sollte als zusätzliche Sicherheit konfiguriert werden, um zu vermeiden, daß bei eventuellen Systemstörungen hohe Telefonkosten entstehen.



Bei diesen Kommandos gibt es Unterschiede zwischen den Herstellern im AT-Kommandosatz (siehe Modemhandbuch).

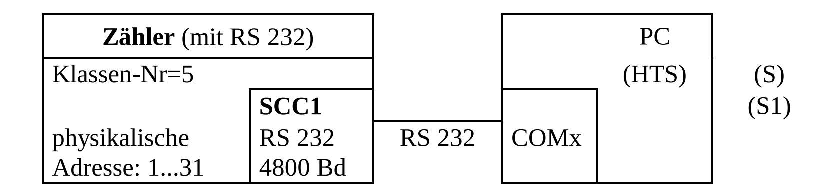
Beispiel:
Initstring für das Modem ***Microlink 2410T2 von ELSA:*** `ATE0V1X0&S0&D2S0=1`

Die Bedeutung im einzelnen:

AT	Präfix Modem AT-Kommando
E0	keine Echo der Kommandos
V1	Antworten im Klartext
X0	Nicht auf Freizeichen warten
&D2	DTR: Modem legt auf, geht in den Kommandomodus und schaltet die Rufaufnahme aus, wenn DTR von ON auf OFF wechselt. Bei Wechsel von OFF auf ON ist die Rufaufnahme wieder möglich.
S0	Register 0 wird auf Wert 1 gesetzt, um die automatische Anschaltung zu aktivieren. Wert muß > 0 sein.

Systemanordnung

Für das korrekte Konfigurieren der Terminal-Klassen und Terminal-Parameter ist es wichtig zu verstehen, welche Zuordnungen vorhanden sind. Beachten Sie, daß der RTC über zwei SCCs (Serial Communication Controller) verfügt, wovon SCC1 immer am PC angeschlossen ist und SCC2 den Zählerbus bedient.
"RS 232" kann ersetzt werden durch "RS232-Modem - - - Modem-RS232".



Zähler (mit RS 485)
Klassen-Nr=5
physikalische Adresse: 1...31
SCC1 RS 485 4800 Bd
RS 485 Bus
RTC
Klassen-Nr=1
SCC2 RS 485
physikalische Adresse=0
SCC1 RS 232 9600Bd
RS 232
COMx
PC (HTS)
(N)
(N1)

Abb. 4: Feste und variable Zuordnungen (= fest, ... variabel)



Die Baudrate am SCC1 der Klassen-Nummer 5 (Hardwaretyp Zähler) darf maximal 4800 betragen, da die Zähler max. 4800 Baud verarbeiten können. Der RTC ist an SCC2 hardwaremäßig auf 4800 Baud eingestellt, während seine SCC1-Schnittstelle auf 9600 voreingestellt ist.

Legen Sie gemäß den Angaben im Kapitel Einführung/Topologie die Netzwerktopologie für jede COM-Schnittstelle fest. Dabei ist nur eine der folgenden Anordnungen möglich:

- Punkt-zu-Punkt (S)
- Punkt-zu-Punkt über Modems (S1)
- 1 bis 31 Zähler über RTC am PC (N)
- 1 bis 31 Zähler über RTC und Modems am PC (N1)

Punkt-zu-Punkt (S), (S1)

Menü Terminal-Konfiguration: *(Speichern nicht vergessen!)*

1. Bestimmen Sie eine logische Terminal-Nummer für den Zähler im Bereich 1-9999. Mit dieser Nummer wird der Zähler eindeutig identifiziert.
2. Bestimmen Sie eine physikalische Adresse für den Zähler im Bereich 1-31. Wählen Sie den korrekten COM-Port aus.
3. Löschen Sie den RTC aus der Terminal-Konfiguration. Dies ist notwendig, damit der Zähler korrekt bedient wird.
4. Wenn der Zähler über Modems verbunden ist, dann müssen Sie noch die vollständige Telefonnummer ohne Sonderzeichen unter Modem/Netzwerk-Adresse eintragen.

Zähler-Hardware:

5. Überprüfen Sie Baudrate, Datenbits und Parität am Zähler und ändern Sie den Funktionscode für die physikalische Adresse gemäß Ihrer vorigen Festlegung. Wie die Funktionscodes programmiert werden, entnehmen Sie bitte der Betriebsanleitung des Zählers.

Menü Terminal-Klassen:

6. Stellen Sie Baudrate, Datenbits und Parität für SCC1 der Klassen-Nummer 5 (Zähler) auf die Werte des Zählers ein. Beachten Sie bitte die maximal mögliche Baudrate des Zählers.

Mehrere Zähler über RTC am PC (N), (N1)

Menü Terminal-Konfiguration: *(Speichern nicht vergessen!)*

1. Bestimmen Sie für jeden Zähler eine eindeutige logische Terminal-Nummer Bereich 1-9999. Mit seiner Nummer wird jeder Zähler eindeutig identifiziert.
2. Bestimmen Sie für jeden Zähler eine eindeutige physikalische Adresse im Bereich 1-31.
3. Überprüfen Sie, ob ein RTC eingetragen ist (Terminal-Klasse ist 0001-RTC). Falls kein Eintrag exisitiert, geben Sie eine neue eindeutige Terminal-Nummer ein und wählen unter Terminal-Klasse den Wert "0001-RTC" aus. Ändern Sie die physikalische Adresse nach 0. Speichern Sie vor dem Verlassen der Maske.
4. Wenn der RTC über Modems verbunden ist, dann müssen Sie noch die vollständige Telefonnummer ohne Sonderzeichen unter Modem/Netzwerk-Adresse eintragen.

Zähler-Hardware:

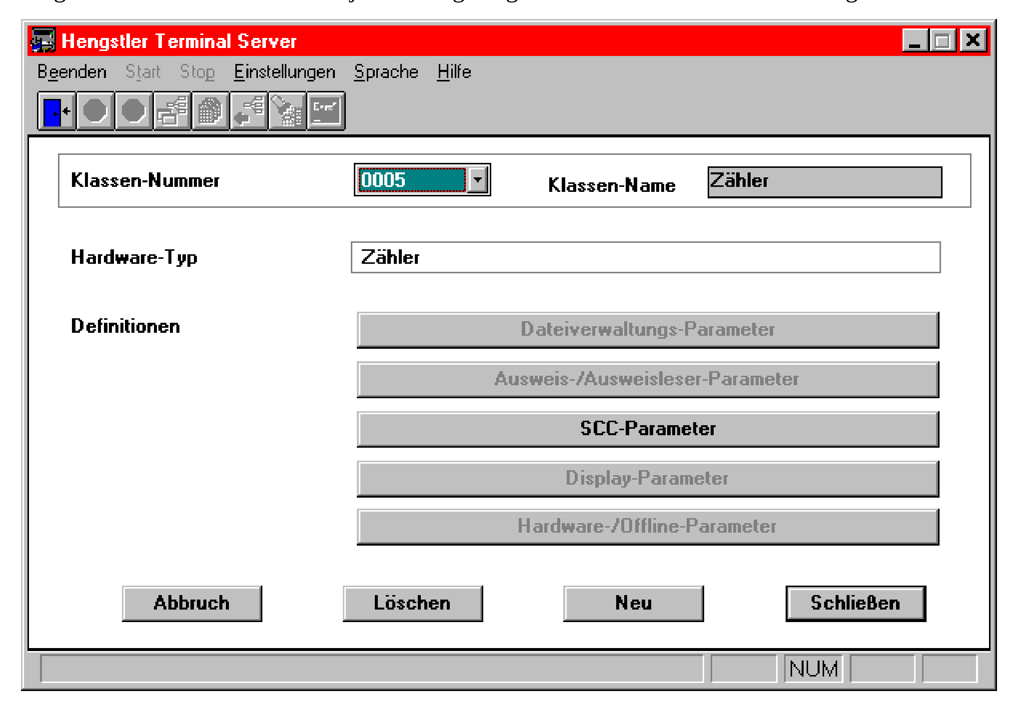
5. Stellen Sie Baudrate, Datenbits und Parität an jedem Zähler gleich ein. Ändern Sie für jeden Zähler den Funktionscode für die physikalische Adresse gemäß Ihren Festlegungen. Wie die Funktionscodes programmiert werden, entnehmen Sie bitte den Betriebsanleitungen der Zähler.

Menü: Terminal-Klassen:

6. Stellen Sie Baudrate, Datenbits und Parität für SCC1 der Klassen-Nummer 5 (Zähler) auf die Werte der Zähler ein. Beachten Sie bitte die maximal mögliche Baudrate der Zähler.
7. Stellen Sie Baudrate, Datenbits und Parität für SCC2 der Klassen-Nummer 1 (RTC) auf die selben Werte ein.
8. Stellen Sie die Baudrate für SCC1 der Klassen-Nummer 1 (RTC) auf den maximalen Wert ein.

Terminal-Klassen

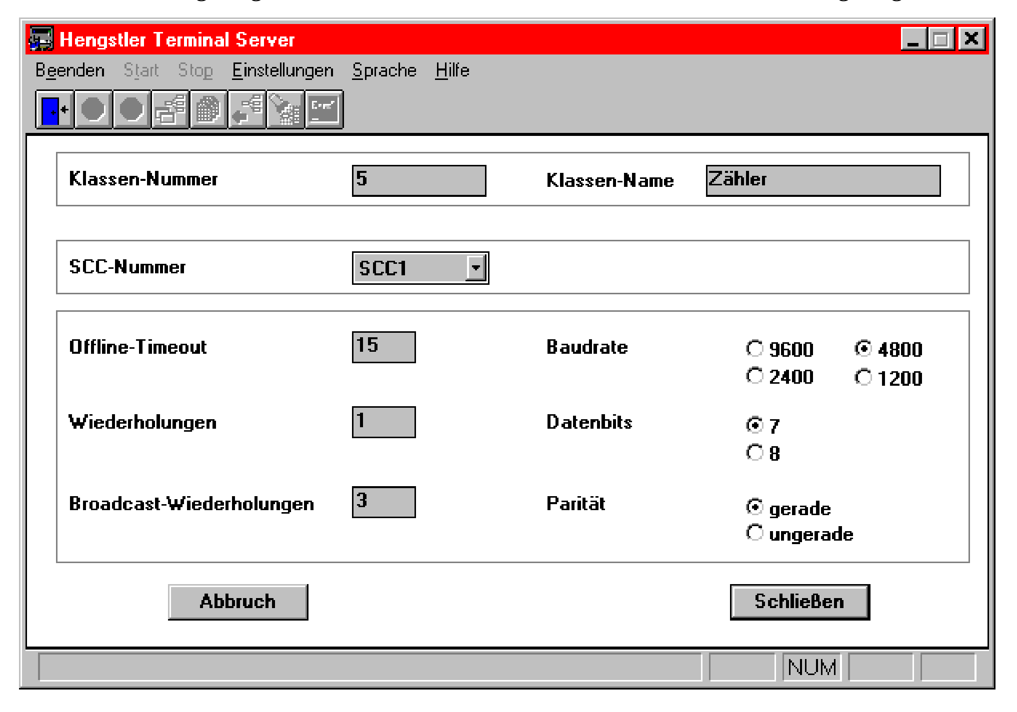
Der Server muß auf Stop stehen, damit diese Maske im Menü Einstellungen aktiviert werden kann. Alle schattiert dargestellten Felder sind für die jeweils angezeigte Geräte-Klasse ohne Bedeutung und können nicht aktiviert werden.



Klassen-Nummer
Die Referenznummer der Klasse. Die Nummer kann nicht geändert werden, da sie nach einer festen Zuordnung vergeben ist.

Klassen-Name
Der Referenzname der Klasse. Der Name kann geändert werden, um z.B. Anpassungen an den Sprachgebrauch vorzunehmen.

SCC-Parameter
Über dieses Feld gelangt man an die Parameter für die Serialschnittstellen der angezeigten Geräte-Klasse:



SCC-Nummer
Zähler besitzen nur eine Serial-Schnittstelle, weshalb hier nur eine Auswahl vorhanden ist.

Offline-Timeout
Hier kann die Offline-Timeout des Terminals in Sekunden festgelegt werden. Antwortet ein konfiguriertes Terminal nach dieser Zeit nicht, wird es als offline eingetragen.

Wiederholungen
Hier kann die Anzahl Wiederholungen bei der Datenübertragung im Fehlerfall festgelegt werden.

Broadcast-Wiederholungen
Hier kann die Anzahl Broadcast-Wiederholungen für die Inter-Terminal-Kommunikation festgelegt werden. Dieser Parameter ist für Zähler und RTC nicht von Bedeutung.

Baudrate
Hier kann die Übertragungsrate festgelegt werden. Beachten Sie die von der Geräteklasse unterstützte Übertragungsrate.

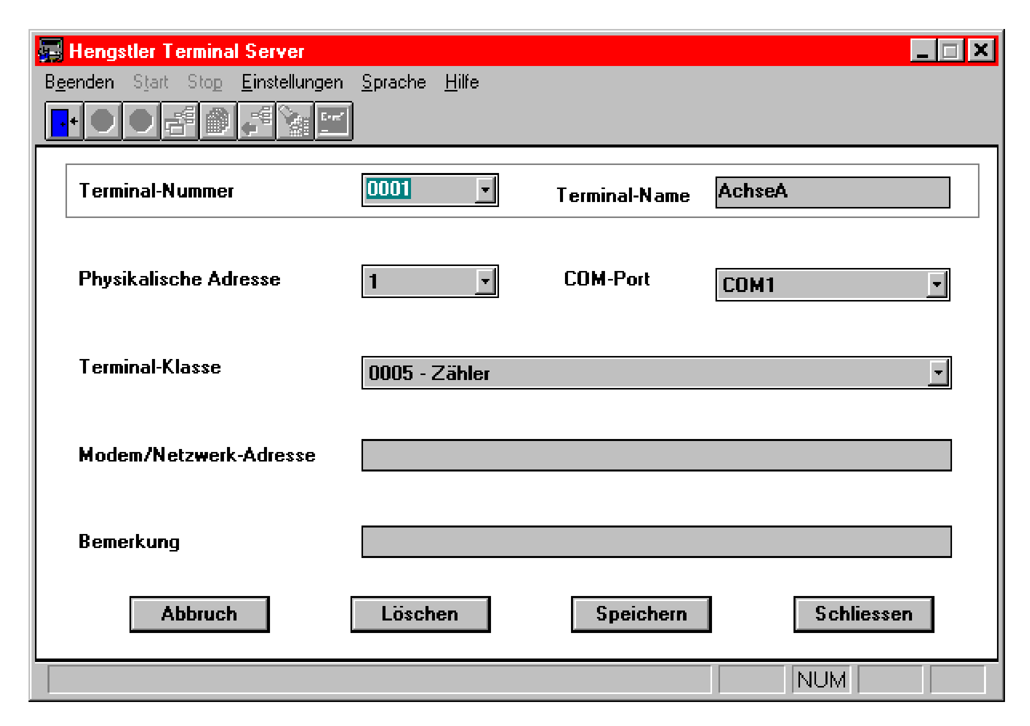
Datenbits / Parität
Anzahl Datenbits und die Parität sollten immer auf 7 und gerade eingestellt bleiben. Diese Einstellungen müssen mit den Funktionscodes der Zähler übereinstimmen.

Terminal-Konfiguration

Der Server muß auf Stop stehen, damit diese Maske im Menü Einstellungen aktiviert werden kann.



Sie müssen Änderungen explizit abspeichern, bevor Sie die Maske verlassen.



Terminal-Nummer
Mit diesem Feld erfolgt die Auswahl der logischen Nummer der vorhandenen Terminals. Ein neues Terminal wird angelegt durch Überschreiben mit einer neuen Nummer. Wertebereich 1-9999.

Terminal-Name
Der Terminal-Name dient der leichteren Zuordnung des Terminals durch den Benutzer. Er wird vom Treiber in der On-/Offline-Liste angezeigt, wenn in den Treiber-Einstellungen die Status-Anzeige auf Namen steht.

Physikalische Adresse
Die physikalische Adresse des Terminals. Jedes Terminal innerhalb eines Busses muß einen eindeutigen Eintrag enthalten. Wertebereich 1-31 für Zähler und 0 für den RTC.

COM-Port
Der serielle COM-Port des PCs, über den das Terminal erreicht werden kann. Bitte verwechseln Sie nicht diesen Wert mit der SCC-Nummer einer Geräte-Klasse. COM-Port entspricht in Abb. 4 dem "x" der Angabe COMx.

Terminal-Klasse
Dieser Eintrag legt fest, welchem Typ von Hardware (Geräte-Klasse) dieses Terminal zugeordnet wird (Zähler oder RTC).

Modem/Netzwerk-Adresse
Tragen Sie hier eine Telefonnummer ein, wenn die Systemanordnung Modems enthält. Siehe auch ***Initialisierung*** und ***Wählen*** in der Maske der Treiber-Einstellungen.

Bemerkung
Dieses Feld steht zu Ihrer freien Verfügung für Kommentare und wird vom Server nicht ausgewertet.

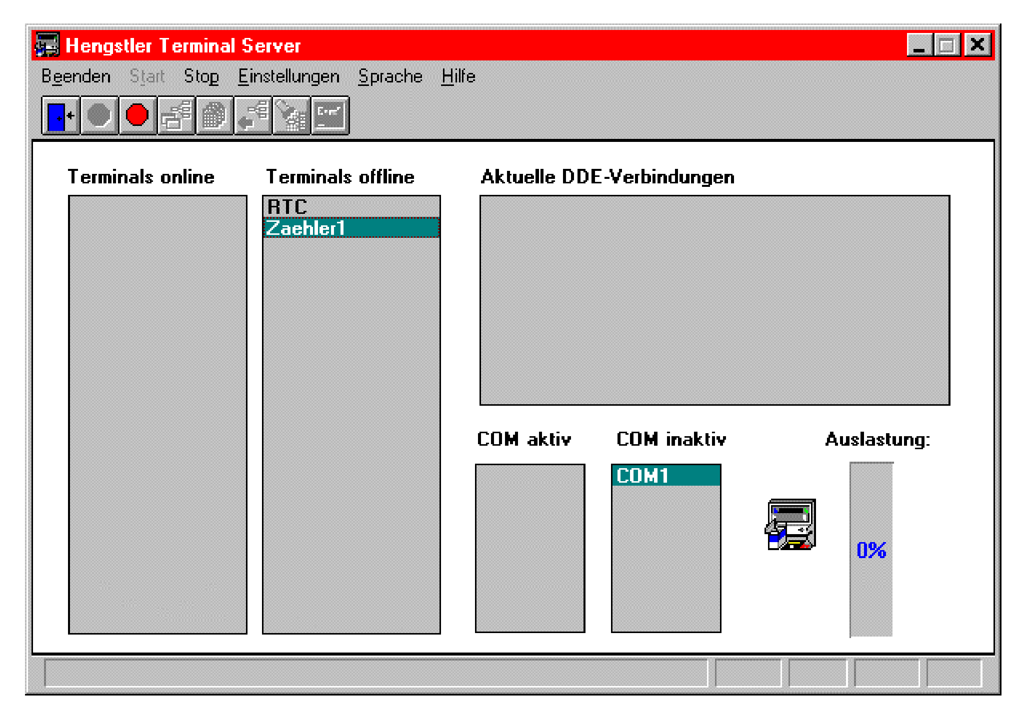
Starten des HTS

Nachdem alle Einstellungen vorgenommen worden sind, können Sie die Hardware anschließen. Vergewissern Sie sich, daß die Leitungslänge des RS232-Kabels im zulässigen Bereich ist. Falls ein RS485-Bus am RTC zum Einsatz

kommt, prüfen Sie bitte, ob ein Busabschluß am äußersten Zähler angeschlossen ist (vgl. Betriebsanleitung/Ergänzung zur Betriebsanleitung des Zählers).



Starten Sie nun den Server entweder durch Neustarten des Programms oder mit dem grünen Startknopf bzw. über das Menü Start.



Dieses Fenster wird nach Start des Servers auf dem Bildschirm angezeigt und bietet einen Überblick über die Grundzustände des Servers.

Terminals online
In diesem Fenster werden alle gültig konfigurierten Terminals angezeigt, die in Verbindung zum Server stehen und deren Initialisierungsdaten dem Server bekannt sind.

Terminals offline
In diesem Fenster werden alle gültig konfigurierten Terminals angezeigt, die zur Zeit keine korrekte Verbindung zum Server haben oder deren Initialisierungsdaten dem Server nicht bekannt sind (z.B. aufgrund eines unbekannten Protokolls). Dazu gehören auch Modem-Terminals zu denen z.Zt. keine Verbindung besteht und Terminals die fehlerhafterweise offline sind. Im Normalzustand sollten hier keine Einträge vorhanden sein.



Doppelklicken auf einen Terminaleintrag öffnet für das entsprechende Terminal ein Übersichtsfenster mit statistischen Angaben. Näheres siehe im Kapitel Diagnose und Fehlerbehandlung.

COM aktiv
Diese Spalte listet die COM-Ports auf, an denen ein gültiges Terminal (RTC oder Zähler) erkannt wurde.

COM inaktiv
Diese Spalte listet die COM-Ports auf, an denen kein gültiges Terminal (RTC oder Zähler) erkannt wurde. Im Normalzustand sollten hier keine Einträge vorhanden sein.



Doppelklicken auf einen COM-Eintrag startet den Line-Monitor für diese Schnittstelle. Näheres siehe im Kapitel Diagnose und Fehlerbehandlung.

Aktuelle DDE-Verbindungen
Diese Fenster zeigt die Fenster-Handles (eine Art fortlaufende von Windows vergebene Numerierung der Anwendungsfenster) und Fenstertitel derjenigen Applikationen, die eine gültige DDE-Verbindung zum Server aufgebaut haben. Jede eintreffende OLE-Anforderung wird fortlaufend durchnumeriert und während ihrer Bearbeitungsdauer angezeigt.

Auslastung

Dieser Graph zeigt die momentane Auslastung des Treibers in Prozent der möglichen Gesamtkapazität.

Integration in Kunden-Anwendung

Die Kommunikation zur Kunden-Anwendung auf dem PC erfolgt über DDE oder OLE.

DDE

HTS ist als Server implementiert, der Client-Programmen Kommunikations-Dienste zur Verfügung stellt. Die gesamte Inter-Prozess-Kommunikation zu diesem Server erfolgt asynchron, da insbesondere bei serieller Datenübertragung und/oder größeren Datenmengen immer mit einer größeren Bearbeitungzeit durch den Server zu rechnen ist.
Als Protokoll zu diesem Server wird aus Gründen des Echtzeitverhaltens, der Prozessorbelastung sowie der Portablität auf Non-Windows-Systeme ein Custom-DDE-Protokoll in der hierin beschriebenen Form verwendet.

Kommunikations-Mechanismen

Grundsätzlich erfolgt bei der Kommunikation ein fester Handshake. Wenn der DDE-Auftrag vom Server entgegen-genommen wurde, erfolgen alle internen Timeout-Überwachungen und Fehlerbehandlungen innerhalb des Servers. Die Client-Anwendung erhält dann entweder das Ergebnis des DDE-Auftrages (mit oder ohne Daten) oder eine qualifizierte Fehlermeldung. Werden Nettodaten geliefert, so sind sie Bestandteil der Quittungs-Message.
DDE-Server bieten ihre Dienste unter Namen an. Damit das Client-Programm die richtigen Dienste auswählt, muß es die Namen kennen und verwenden.



Service: "Hengstler Terminal Server"
Topic: "All Items"
Der Datenaustausch erfolgt ausschließlich asynchron und binär. Die Kommandostruktur muß für DATA Request immer verwendet werden.

Schematische Darstellung der Server-Kommunikation

Applikation		HTS
DDE Connect	⇒	Client akzeptieren
Verbindung zu HTS steht	⇐	DDE Acknowledge
DDE Data Request	Kommandostruktur ⇒	Kommunikation mit gewünschtem Zähler. Zähler auslesen oder schreiben
Nettodaten verarbeiten	⇐ Kommandostruktur	DDE Data
DDE Disconnect	⇒	Client austragen
Verbindung zu HTS aufgelöst	⇐	DDE Disconnect

Aufbau der Kommandostruktur

Bei der Kommandostruktur handelt es sich um eine C-Struktur die folgendermaßen deklariert ist:

```
typedef struct tp3_command_s {
   long size;                // Größe der Struktur
   long pid;                 // Window-Handle der Applikation
   long logtnr;              // logische Terminalnummer (0000-9999)
   long jobid;               // Auftagszähler
   long service;             // Dienst (siehe unten)
   long iodesc;              // File/Devicenummer für Ein-/Ausgabe
   long datatype;            // Datenart
   long records;             // Anzahl Records
   long response;            // mit/ohne Quitttung
   long error;               // Fehlermerker
   char errortext[ERRTEXTLEN]; // Fehlertext
   char filename[MAXPATH+1];  // Dateiname
}tp3_command_struct;
```

Die Kommando-Struktur ist Header und somit Bestandteil jedes ausgetauschten DDE-Pakets. Werden Daten mitgeliefert, so wird das DDE-Paket dementsprechend größer definiert und die Daten in der unten dokumentierten Form an die Struktur angehängt.

Achtung: wie in DDE üblich, wird die empfangene Datenstruktur immer vom Empfänger deallokiert!



Das Vorhandensein von Daten wird über den Größeneintrag **size** in der Kommando-Struktur festgestellt.
databytes = size - sizeof(tp3_command_s);

size - Größe
Dieser Eintrag beinhaltet die Gesamtgröße des DDE-Pakets, einschließlich eventuell enthaltener Parameter oder Daten, d.h. die allokierte Größe des Gesamtpakets in Bytes. Dieser Eintrag ist obligatorisch.

pid - Prozeß-ID
Im Feld Prozeß-ID muß das Window-Handle der Client-Applikation eingetragen werden. Der Eintrag ist obligatorisch.

logtnr - Logische Terminal-Nummer
Alle Kommunikation zum Terminal erfolgt über die logische Terminal-Nummer. Diese wird vom HTS auf die physikalische Adresse entsprechend der Terminal-Konfiguration umgesetzt, sowie die Art der Kommunikation bestimmt (COM1-X, Modem, ...). Der Eintrag ist obligatorisch.

jobid - Auftragszähler
Der Auftragszähler dient in der Applikation zur Zuordnung einer Quittung zum Auftrag. Der Treiber trägt diesen Wert im Quittungspaket entsprechend wieder ein. Dieser Eintrag ist optional.

service - Dienst
Dieser Eintrag beschreibt die Art des vom Treiber auszuführenden Auftrags. Verwenden Sie die Definitionen aus der Headerdatei ***types.h***. Dieser Eintrag ist obligatorisch.

iodesc - Datei-/Servicenummer
Dieser Eintrag beschreibt die Datei- oder in Verbindung mit einem Device-Kommandos eine Devicenummer welche über den Dienst bearbeitet werden soll. Dieser Eintrag ist abhängig vom verwendeten Dienst. Die Definitionen befinden sich in ***types.h***. Für die Services bei den Device-Kommandos ergibt sich der Eintrag aus: Devicenummer * 1000 + Register-Nr.

datatype - Nettodatenstruktur
reserviert, z.Zt. nicht verwendet

records - Anzahl Records
Dieser Eintrag beschreibt die Anzahl von Datensätzen. Dieser Eintrag ist abhängig vom verwendeten Dienst und dort detailliert beschrieben.

response - Antworttyp
Dieser Eintrag gibt an, um welchen Pakettyp es sich handelt. Folgende Einträge sind möglich (vgl. ***types.h***):

```
NO_RESPONSE           -> Das Paket soll nicht quittiert werden
SEND_RESPONSE         -> Quittungspaket wird erwartet
IS_RESPONSE           -> Es handelt sich um ein Quittungspaket
```

Dieser Eintrag ist obligatorisch.

error - Quittungwert
Beim Quittungspaket wird hier ein eventuell aufgetretener Fehler eingetragen. Es sind viele verschiedene Werte möglich. Für eine genaue Beschreibung und Zuordnung der einzelnen Werte siehe Detail-Dokumentation im entsprechenden Headerfile. Bei Auftragspaketen muß dieser Wert 0 sein. NOERROR ist der Eintrage bei fehlerfreiem Auftrag.

errortext - Fehlermeldung im Klartext
Bei Quittungspaketen wird in diesem Feld eine Fehlermeldung als Kurztext eingetragen, der von der Applikation verwendet werden kann. Es wird die im Treiber konfigurierte Sprache verwendet. In Auftragspaketen wird dieses Feld nicht verwendet.

filename - Dateiname
reserviert, z.Zt. nicht verwendet

Datenaustausch mit Zählern

Bei der Kommunikation mit Zählern sind nur Lese- und Schreibzugriffe auf sogenannte **Devices** (engl.: Geräte) möglich. Es handelt sich dabei um festgelegte Bereiche von Nummern, die für bestimmte Hardwaretypen der verschiedenen Terminals festgelegt sind.



Für Zähler ist die Devicenummer **73**.
Die einzelnen Register sind

Nr.	Beschreibung	Zugriff
000	Zählerstand	(rd, wr, rd&cl)
001	Vorwahl 1	(rd, wr, rd&cl)
002	Vorwahl 2	(rd, wr, rd&cl)
003	Kettenwert (Chain)	(rd, rd&cl)
004	Setzwert	(rd, wr, rd&cl)
005	Prescaler	(rd)
006	Dezimalpunkt	(rd)
007	Version	(rd)



Das Feld iodesc in der Kommandostruktur errechnet sich wie folgt:
iodesc = 1000*Devicenummer + Registernummer

Für Zähler (device 73) ergibt sich folgende Übersicht:

Dienst	Register	Reg. -Nr.	Eintrag in ***service***	Eintrag in ***iodesc***	Datenformat* (immer ASCII)
Lesen	Zählerstand	000	READ_DEVICE	73000	8 Zeichen, linksbündig
Lesen	Vorwahl1	001	READ_DEVICE	73001	8 Zeichen, linksbündig
Lesen	Vorwahl 2	002	READ_DEVICE	73002	8 Zeichen, linksbündig
Lesen	Kettenwert	003	READ_DEVICE	73003	8 Zeichen, linksbündig
Lesen	Setzwert	004	READ_DEVICE	73004	8 Zeichen, linksbündig
Lesen	Prescaler	005	READ_DEVICE	73005	8 Zeichen, linksbündig
Lesen	Dezimalpunkt	006	READ_DEVICE	73006	2 Zeichen, linksbündig, 0=kein Dez.Pkt.
Lesen	Version	007	READ_DEVICE	73007	8 Zeichen, Text
Lesen&Reset	Zählerstand	000	RD_CL_DEVICE	73000	8 Zeichen, linksbündig
Lesen&Reset	Vorwahl 1	001	RD_CL_DEVICE	73001	8 Zeichen, linksbündig
Lesen&Reset	Vorwahl 2	002	RD_CL_DEVICE	73002	8 Zeichen, linksbündig
Lesen&Reset	Kettenwert	003	RD_CL_DEVICE	73003	8 Zeichen, linksbündig
Lesen&Reset	Setzwert	004	RD_CL_DEVICE	73004	8 Zeichen, linksbündig
Schreiben	Zählerstand	000	WRITE_DEVICE	73000	8 Zeichen, linksbündig
Schreiben	Vorwahl1	001	WRITE_DEVICE	73001	8 Zeichen, linksbündig
Schreiben	Vorwahl 2	002	WRITE_DEVICE	73002	8 Zeichen, linksbündig
Schreiben	Setzwert	004	WRITE_DEVICE	73004	8 Zeichen, linksbündig
Schreiben	Dezimalpunkt	006	WRITE_DEVICE	73006	2 Zeichen, linksbündig, 0=kein Dez.Pkt.
Schreiben	Version	007	WRITE_DEVICE	73007	8 Zeichen, Text

* rechts mit Leerzeichen aufgefüllt.

Beispiel: Schreiben Zählerstand "1234,98". Wenn der Dez.Pkt am Zähler auf 1 steht, also XXX.X, wird nur der relevante Wert 1234,9 übernommen.



Unmittelbar im Anschluß an die Kommandostruktur wird der Datenbereich für die Registerdaten angehängt. Bitte beachten Sie, daß keine Nullterminierung der Strings erfolgt. Die Angabe in ***size*** spiegelt genau die gesamte Anzahl Bytes (Kommandostruktur + allokierter Datenbereich) wieder. HTS nimmt keine Format- oder Längenprüfung vor. Falsche Formate führen zu einer negativen Quittung beim Zähler mit einem entsprechende Eintrag im Feld ***error***.



Es findet keine Null-Terminierung der Nettodaten statt.

OLE

Solange der HTS eine OLE-Anfrage bearbeitet, zeigt er dies im Hauptfenster unter "aktive DDE-Verbindungen" durch eine fortlaufende Numerierung des Vorgangs. Anhand von Beispielen sehen Sie die Implentierung der OLE-Schnittstelle in Ihre Anwendungen.

Excel und Visual Basic

Die einfachste Implementierung läßt sich über Visual Basic realisieren, was auch Bestandteil von Microsoft Excel ist. Folgendes Beispiel liest aus dem Zähler mit der logischen Terminalnummer 25 den Zählerstand aus bzw. setzt das Vorwahl1-Register.

```
' Logische Zähleradresse
Konst ZählerAdresse = 25
' Die einzelnen Register eines Zählers
Konst Zählerstand = 0
Konst Vorwahl1 = 1
Konst Vorwahl2 = 2
Konst Chain = 3
' Definition der Übergabevariablen
Dim Buffer Als Einfach

' Der Zählerstand wird ausgelesen und in die Tabelle1 geschrieben.
Sub Read_Counter()
    Setze Hts = HoleObjekt(Klasse:="Hengstler.TerminalServer.10")
    Result = Hts.ReadRegister(ZählerAdresse; Zählerstand)
    BlattListe("Tabelle1").ZelleListe(6; 2).Wert = Result
Ende Sub

Sub Write_Preset()
    Buffer = BlattListe("Tabelle1").ZelleListe(2; 2).Wert
    Setze Hts = HoleObjekt(Klasse:="Hengstler.TerminalServer.10")
    Result = Hts.WriteRegister((ZählerAdresse; Vorwahl1; Buffer)
Ende Sub

Sub Read_Clear_Counter()
    Setze Hts = HoleObjekt(Klasse:="Hengstler.TerminalServer.10")
    Result = Hts.ClearRegister(ZählerAdresse; Zählerstand)
    BlattListe("Tabelle1").ZelleListe(6; 2).Wert = Result
Ende Sub
```

Beispiel: OLE mit Visual Basic

C++ Anwendungen

Für dieses Beispiel liegt ***MS Visual C++ Design Studio*** zugrunde. Es basiert auf der sample application **`...\samples\ole\calcdriv`**.

Die Klassendefinitionen für "Hengstler Terminal Server" werden automatisch generiert mit folgendem Ablauf:

1. HTS beenden
2. Auf Kommandozeile (Datei Ausführen bzw. Start-Ausführen) **hentmsrv.exe /typelib**
   Damit erzeugen Sie eine Object Library datei ***hentmsrv.olb*** im Verzeichnis des HTS
3. Erzeugen Sie mit dem Class Wizard eine neue Klasse, abgeleitet aus ***hentmsrv.olb***
4. ***class HengstlerTerminalServer*** kann nun eingebaut werden

Auszüge aus Programmdateien veranschaulichen die wichtigsten Passagen.

hentmsrv.h:

```
// Machine generated IDispatch wrapper class(es) created with ClassWizard
/////////////////////////////////////////////////////////////////////////////
// HengstlerTerminalServer wrapper class

class HengstlerTerminalServer : public COleDispatchDriver
{
public:
	HengstlerTerminalServer() {}		// Calls COleDispatchDriver default constructor
	HengstlerTerminalServer(LPDISPATCH pDispatch) : COleDispatchDriver(pDispatch) {}
	HengstlerTerminalServer(const HengstlerTerminalServer& dispatchSrc) :
COleDispatchDriver(dispatchSrc) {}

// Attributes
public:

// Operations
public:
	CString ReadRegister(long File, long Number);
	CString ClearRegister(long File, long Number);
	CString WriteRegister(long File, long Number, LPCTSTR Value);
	CString ReadDevice(long File, long Number);
	CString WriteDevice(long File, long Number, LPCTSTR Value);
	CString ReadFile(long File, long Number);
	CString WriteFile(long File, long Number, LPCTSTR Value);
};
```

Initialisierung von OLE:

```
BOOL CCalcDrivApp::InitInstance()
{
#ifdef _DEBUG
    #ifndef _MAC
          // turn on extra memory tracking
	afxMemDF |= checkAlwaysMemDF;
    #endif
#endif
	// Initialize OLE 2.0 libraries
	if (!AfxOleInit())
	{
		AfxMessageBox(IDP_OLE_INIT_FAILED);
		return FALSE;
	}

	// Standard initialization
	// If you are not using these features and wish to reduce the size
	//  of your final executable, you should remove from the following
	//  the specific initialization routines you do not need.

	Enable3dControls();    // Use 3d controls in dialogs

	// Simple application that simply invokes a dialog
	CDriverDlg dlg;
	m_pMainWnd = &dlg;
	dlg.DoModal();

	return FALSE;   // don't run after the dialog is done
}
```

hentmsrv.cpp:

```
// Machine generated IDispatch wrapper class(es) created with ClassWizard
```

```
// To redo creation create the HTS-OLE typelib by calling
// HENTMSRV.EXE /TYPELIB
// The file HENTMSRV.OLB is automatically generated for you
// In the class wizard choose Add Class and From an OLE TypeLib
// A dialog box appears: Choose the previously generated HENTMSRV.OLB
// This file and the header file will then automatically be generated for you
// You can then directly use all HengstlerTerminalServer member functions

#include "stdafx.h"
#include "hentmsrv.h"

#ifdef _DEBUG
#define new DEBUG_NEW
#undef THIS_FILE
static char THIS_FILE[] = __FILE__;
#endif



/////////////////////////////////////////////////////////////////////////////
// HengstlerTerminalServer properties

/////////////////////////////////////////////////////////////////////////////
// HengstlerTerminalServer operations

CString HengstlerTerminalServer::ReadRegister(long File, long Number)
{
	CString result;
	static BYTE parms[] =
		VTS_I4 VTS_I4;
	InvokeHelper(0x1, DISPATCH_METHOD, VT_BSTR, (void*)&result, parms,
		File, Number);
	return result;
}

CString HengstlerTerminalServer::ClearRegister(long File, long Number)
{
	CString result;
	static BYTE parms[] =
		VTS_I4 VTS_I4;
	InvokeHelper(0x1, DISPATCH_METHOD, VT_BSTR, (void*)&result, parms,
		File, Number);
	return result;
}


CString HengstlerTerminalServer::WriteRegister(long File, long Number, LPCTSTR Value)
{
	CString result;
	static BYTE parms[] =
		VTS_I4 VTS_I4 VTS_BSTR;
	InvokeHelper(0x2, DISPATCH_METHOD, VT_BSTR, (void*)&result, parms,
		File, Number, Value);
	return result;
}

CString HengstlerTerminalServer::ReadDevice(long File, long Number)
{
	CString result;
	static BYTE parms[] =
		VTS_I4 VTS_I4;
	InvokeHelper(0x3, DISPATCH_METHOD, VT_BSTR, (void*)&result, parms,
		File, Number);
	return result;
}

CString HengstlerTerminalServer::WriteDevice(long File, long Number, LPCTSTR Value)
{
	CString result;
	static BYTE parms[] =
		VTS_I4 VTS_I4 VTS_BSTR;
	InvokeHelper(0x4, DISPATCH_METHOD, VT_BSTR, (void*)&result, parms,
		File, Number, Value);
	return result;
}

CString HengstlerTerminalServer::ReadFile(long File, long Number)
{
	CString result;
	static BYTE parms[] =
		VTS_I4 VTS_I4;
	InvokeHelper(0x5, DISPATCH_METHOD, VT_BSTR, (void*)&result, parms,
		File, Number);
	return result;
}
```

```
CString HengstlerTerminalServer::WriteFile(long File, long Number, LPCTSTR Value)
{
	CString result;
	static BYTE parms[] =
		VTS_I4 VTS_I4 VTS_BSTR;
	InvokeHelper(0x6, DISPATCH_METHOD, VT_BSTR, (void*)&result, parms,
		File, Number, Value);
	return result;
}
```

hconnect.cpp:

```
// This example is taken from the Microsoft example CALCDRIV
// and has been modified for use with Hengstler Terminal Server
BOOL CDriverDlg::OnInitDialog()
{
	CDialog::OnInitDialog();

	// create the hts object that we'll drive through OLE automation
	COleException e;
	CLSID clsid;
	if (CLSIDFromProgID(OLESTR("Hengstler.TerminalServer.10"), &clsid) !=
NOERROR)
	{
		AfxMessageBox(IDP_UNABLE_TO_CREATE);
		EndDialog(IDABORT);
		return FALSE;
	}

	// try to get the active hts before creating a new one
	LPUNKNOWN lpUnk;
	LPDISPATCH lpDispatch;
	if (GetActiveObject(clsid, NULL, &lpUnk) == NOERROR)
	{
		HRESULT hr = lpUnk->QueryInterface(IID_IDispatch,
			(LPVOID*)&lpDispatch);
		lpUnk->Release();
		if (hr == NOERROR)
			m_hts.AttachDispatch(lpDispatch, TRUE);
	}

	// if not dispatch ptr attached yet, need to create one
	if (m_hts.m_lpDispatch == NULL &&
		!m_hts.CreateDispatch(clsid, &e))
	{
		AfxMessageBox(IDP_UNABLE_TO_CREATE);
		EndDialog(IDABORT);
		return FALSE;
	}

	// read a counter register to test and see if it works

	CString result = m_hts.ReadRegister(1,1);

	// examine result in the debugger

	return TRUE;  // return TRUE  unless you set the focus to a control
}
```

Geschwindigkeit erhöhen

Um den Overhead des Öffnen Device und Schließen Device im HTS bei jeder DDE- oder OLE-Anfrage zu unterdrücken, kann folgender Trick angewendet werden. Dazu fügen Sie manuell (HTS zuvor beenden) in hentmsrv.ini die nachfolgenden Zeilen am Ende ein:

hentmsrv.ini:

```
[MultiTasking]
DirectAccess=1
```



Diese Änderung ist nur sinnvoll, wenn HTS nur von einer Anwendung exklusiv verwendet wird. Bitte beachten Sie die Großkleinschreibung unbedingt.

Diagnose und Fehlerbehandlung

Dieses Kapitel behandelt hauptsächlich Dinge, die den Servicefall betreffen.

Einstellungen im Überblick

Dieses Kapitel dient nur zur Anschauung oder als Hilfe in besonderen Fällen, z.B. zu Servicezwecken. Die INI-Dateien werden von HTS manipuliert und dürfen nicht gelöscht oder verändert werden.

hentmsrv.ini: allgemeine Treibereinstellungen
vt3tm.ini: Liste aller konfigurierten Terminals
vt3c0000.ini: Definitionen der Terminal-Klasse 0 (RTC)
vt3c0005.ini: Definitionen der Terminal-Klasse 5 (Zähler)



Manuelle Änderungen in den INI-Dateien sind nicht zulässig und können zu unvorhersehbaren Auswirkungen führen! Groß- und Kleinschreibung wird unterschieden.

hentmsrv.ini

```
[Global]
Version=1
HelpLevel=2
Language=0
WindowMode=0
LogfileSize=300000
StatusMode=0

[Modem]
InitString=AT&FB0X3&S1\N1\C2%C0S0=0L1&W0Z0
RInitString=AT&F5E0M0X3&R1&S1\N1\C2%C0S0=1&W0Z0
DialPrefix=ATDT

[Timeouts]
HelpTimeout=1000
OfflineTimeout=30
JobTimeout=10
IdleTimeout=0
TerminalListTimeout=60
ReinitTimeout=60
ModemConnectTimeout=60
ModemReuseTimeout=1
ItsByteTimeout=3
ItsEnqTimeout=20
ItsCmdTimeout=4

[COM1]
Device=0
[COM2]
Device=0
...
[COM9]
Device=0
```

vt3tm.ini

```
// Terminal-Definition
// Created by Hengstler Terminal Server
// Last Change: 28.05.1997 18:19:05

[TM0000]
Name=RTC
PhysicalTerminalAdress=0
TMClass=0
COM-Port=1
RemoteAccess=
Remark=

[TM0001]
Name=Zaehler1
PhysicalTerminalAdress=1
TMClass=5
COM-Port=1
RemoteAccess=
Remark=
```

vt3c0000.ini

Klassendefinition für RTC, Terminalklasse 0

```
// Class-Definition
// Created by Hengstler Terminal Server
// Last Change: 28.10.1996 15:25:24

[Version]
Version=1

[Device]
Name=RTC
DeviceClass=3
DeviceType=3

[SCC1]
Baudrate=5
DataBitsCount=7
Parity=0
OfflineTimeout=5
Repetitions=3
BroadcastRepetitions=2
SendBeginDelay=0
SendEndDelay=0

[SCC2]
Baudrate=4
DataBitsCount=7
Parity=0
OfflineTimeout=5
Repetitions=3
BroadcastRepetitions=2
SendBeginDelay=0
SendEndDelay=0
```

vt3c0005.ini

Klassendefinition für Zähler, Terminalklasse 5

```
// Class-Definition
// Created by Hengstler Terminal Server
// Last Change: 03.02.1997 15:15:10

[Version]
Version=1

[Device]
Name=Zähler
DeviceClass=1
DeviceType=5

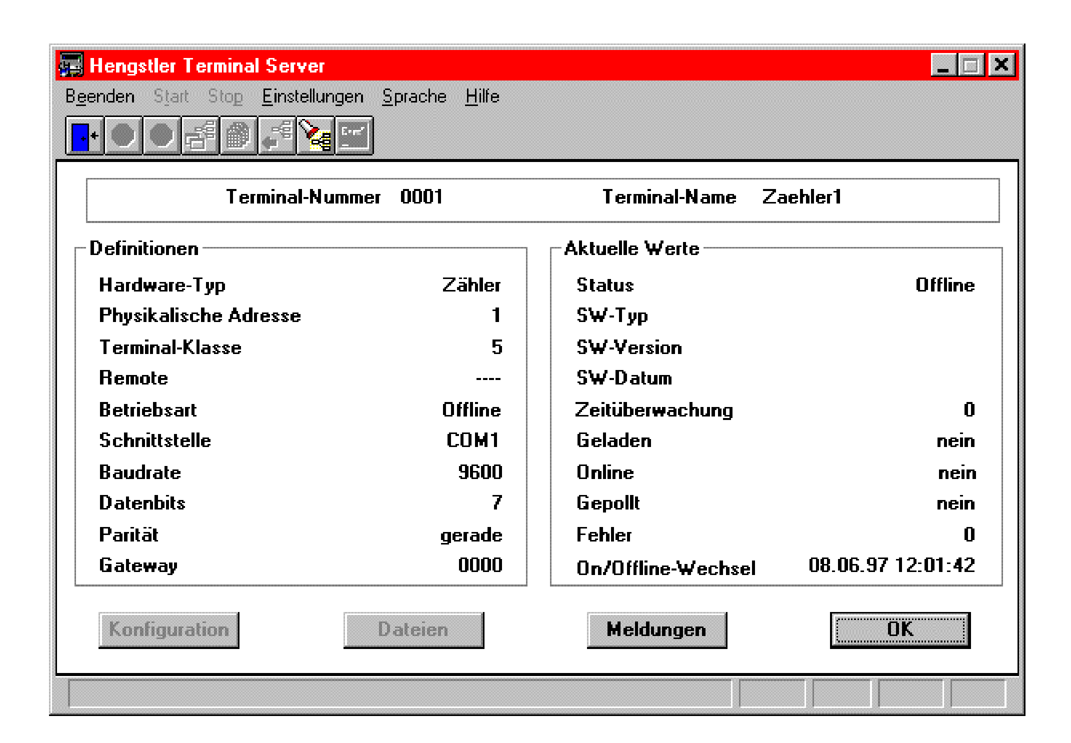
[SCC1]
Baudrate=4
DataBitsCount=7
Parity=0
OfflineTimeout=15
Repetitions=1
BroadcastRepetitions=3
SendBeginDelay=0
SendEndDelay=0
```

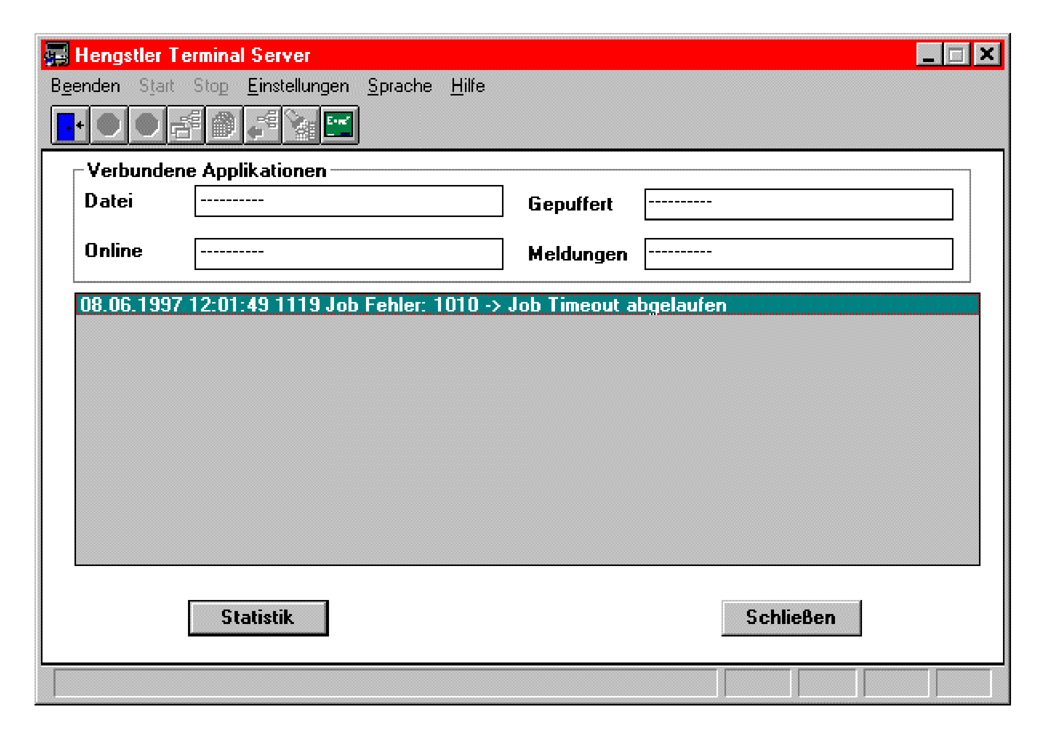
Statistik

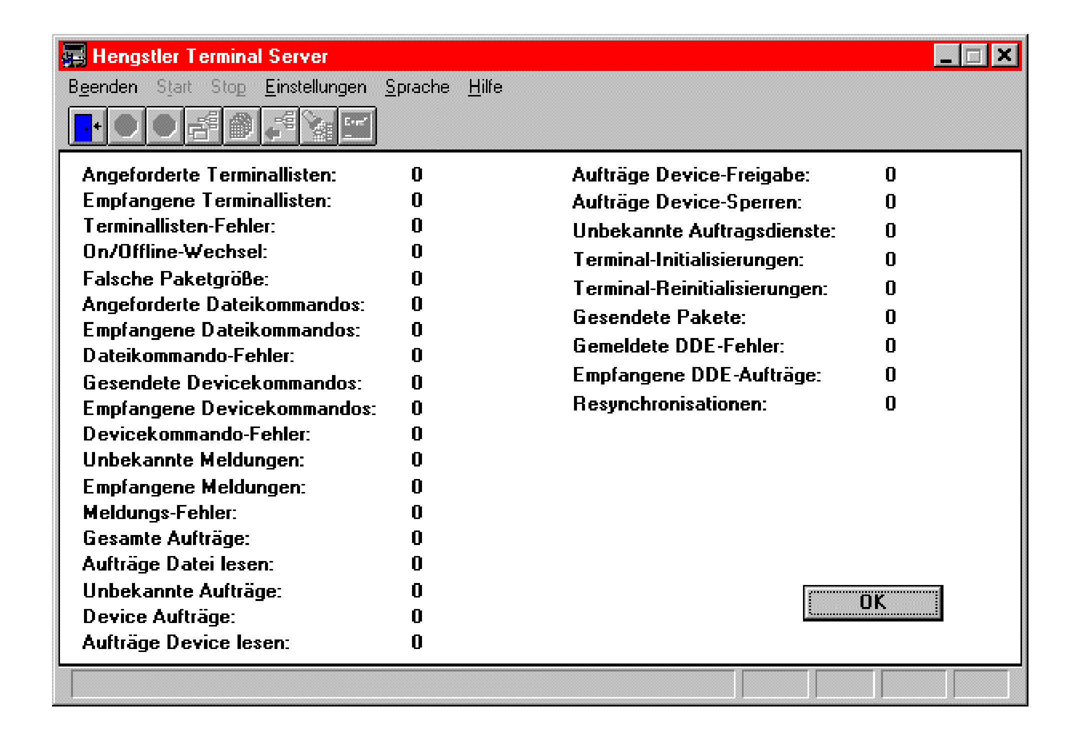
Durch Doppelklicken auf einen Terminal-Eintrag in einer der Spalten *Terminals online* oder *Terminals offline* des Hauptfensters erhalten Sie eine Übersicht des jeweiligen Gerätes. Da es sich hier um Servicefunktionen handelt, soll hier nicht näher auf die einzelnen Masken eingegangen werden.



Die Angabe COMx unter Schnittstelle gibt immer den COM-Port des PCs an, über den das Gerät direkt oder indirekt (via RTC) erreicht wird.

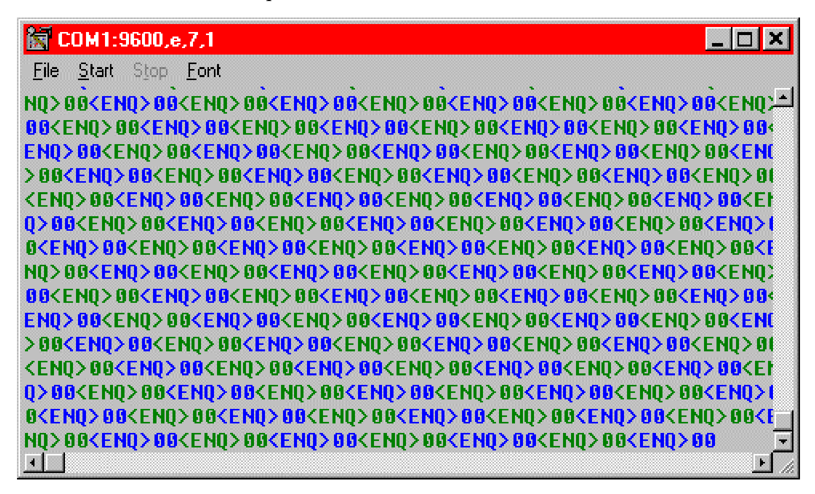






Line Monitor

Durch Doppelklicken auf einen COM-Eintrag wird für die entsprechende Schnittstelle ein Fenster geöffnet, das den Datenverkehr sichtbar macht. Dies ist jedoch nur für Servicezwecke erforderlich, weshalb hier nicht näher darauf eingegangen werden soll. Über Stop und Start können Sie die Aufzeichnung anhalten bzw. fortsetzen, z.B. wenn Sie eine bestimmte Datensequenz näher untersuchen wollen.



Modembetrieb

Es folgt eine Kurzbeschreibung des Ablaufes bei Verbindungsauf- und abbau bei Modembetrieb.

Verbindungsaufbau

Der PC aktiviert die Leitung DTR und das Modem beantwortet dies mit dem Aktivieren der Leitung DSR. Dieser Handshake bleibt während der ganzen Übertragung bestehen. Der Initialisierungsstring kann nun gesendet werden.Nach der Initialisierung sendet der PC die Rufnummer zum Modem. Die Rufnummer wird unter Berücksichtigung der Verbindungsart und dem AT-Präfix zum Modem gesendet. Mit dem Absenden der Rufnummer wird die Modemtimeout gestartet. Antwortet das Modem mit einem CONNECT, so steht die Verbindung zum Terminal. Der Pollbetrieb wird jetzt aufgenommen.Antwortet das Modem nicht mit einem CONNECT oder die Anwortet bleibt innerhalb der Timeout aus, so wird der Verbindungsaufbau mit entsprechender Meldung abgebrochen.

Verbindungsabbau

Sind die Terminaldaten ausgetauscht, so muß die Verbindung wieder ordentlich abgebaut werden. Um dem Terminal mitzuteilen, daß kein Datenaustausch mehr stattfindet, wird ein EOT zum Terminal gesendet. Das Terminal legt nun seinerseits den "Hörer" auf und der PC auf der Gegenseite auch. Der PC deaktiviert die DTR-Leitung, das Modem deaktiviert seinerseits die DSR-Leitung. Nach 1,5 sek. wird ein +++ (Umschalten in den Kommandomode) mit anschliessendem ATH zum Modem geschickt, um per Kommando aufzulegen. Auf der Terminalseite geschieht dies zeitgleich.

Fehlermeldungen

Die häufigsten Meldungen:

Fehler 10 beim Öffnen von
COMx !
...
Terminal y ausgesetzt
COMx nicht ansprechbar !

> die genannte Schnittstelle COMx ist durch ein anderes Programm in Windows belegt. HTS bedient keinen der auf COMx zugeordneten Zähler
* Entsprechendes Programm beenden bzw. eine freie Schnittstelle verwenden

Doppelte Terminalnummer
Terminal xx
Terminal ausgesetzt !

> Sie verwenden die gleiche physikalische Zähleradresse für den Zähler mit der Terminal-Nummer xx. HTS bedient den genannten Zähler nicht.
* HTS auf Stop; Einstellungen->Terminal-Konfiguration: physikalische Adresse ändern und speichern

RTC nicht definiert für NULL
Schnittstelle ausgesetzt !
...
Terminal 1 ausgesetzt
Unknown nicht ansprechbar !
... für alle Zähler

> Sie haben mehr als einen Zähler für eine bestimmte COM-Schnittstelle angelegt
* HTS auf Stop; Einstellungen->Terminal-Konfiguration: Alle Zähler, außer einen, für diese COM löschen